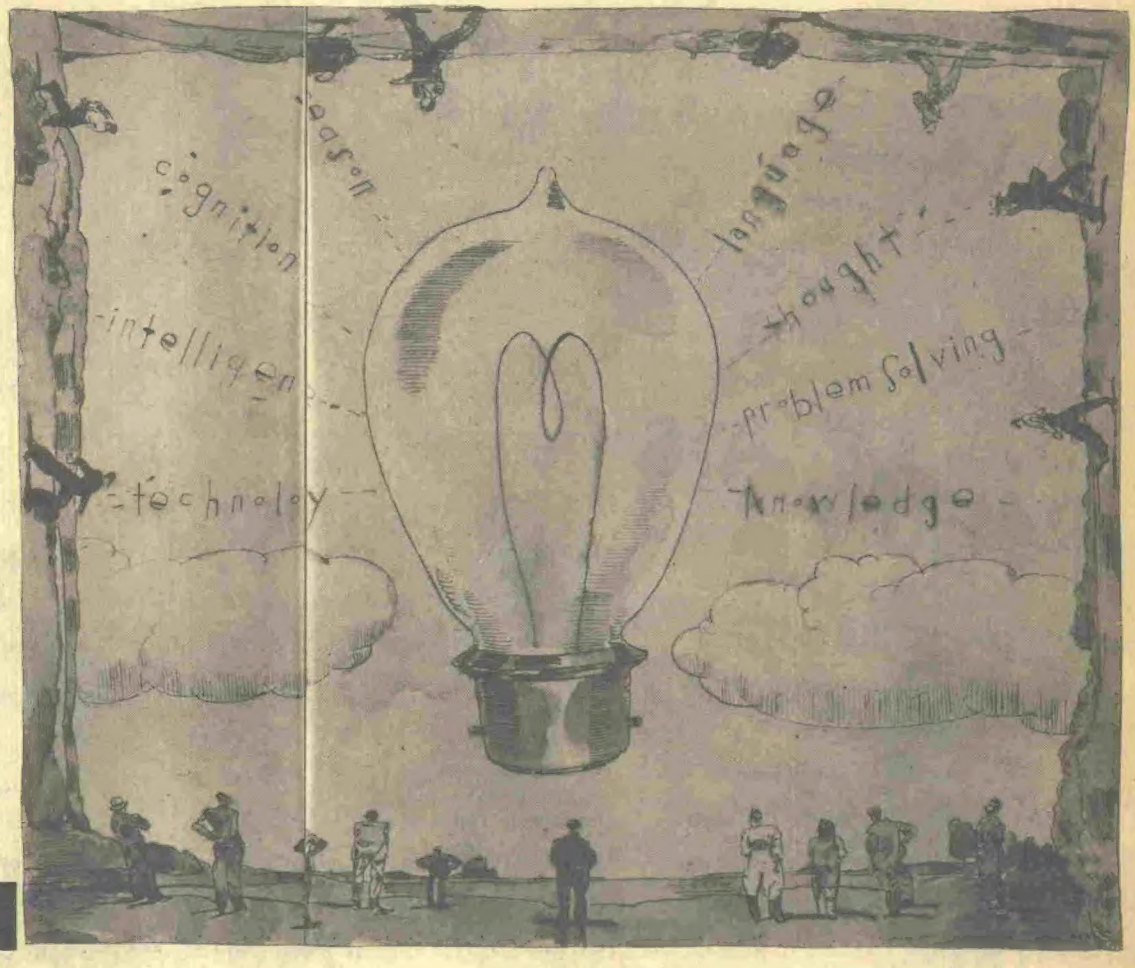
# ย้อนรอยอดีต
# โปรแกรมชาญฉลาด (Intelligent Program)
## ก่อนจะถึงยุครุ่งโรจน์

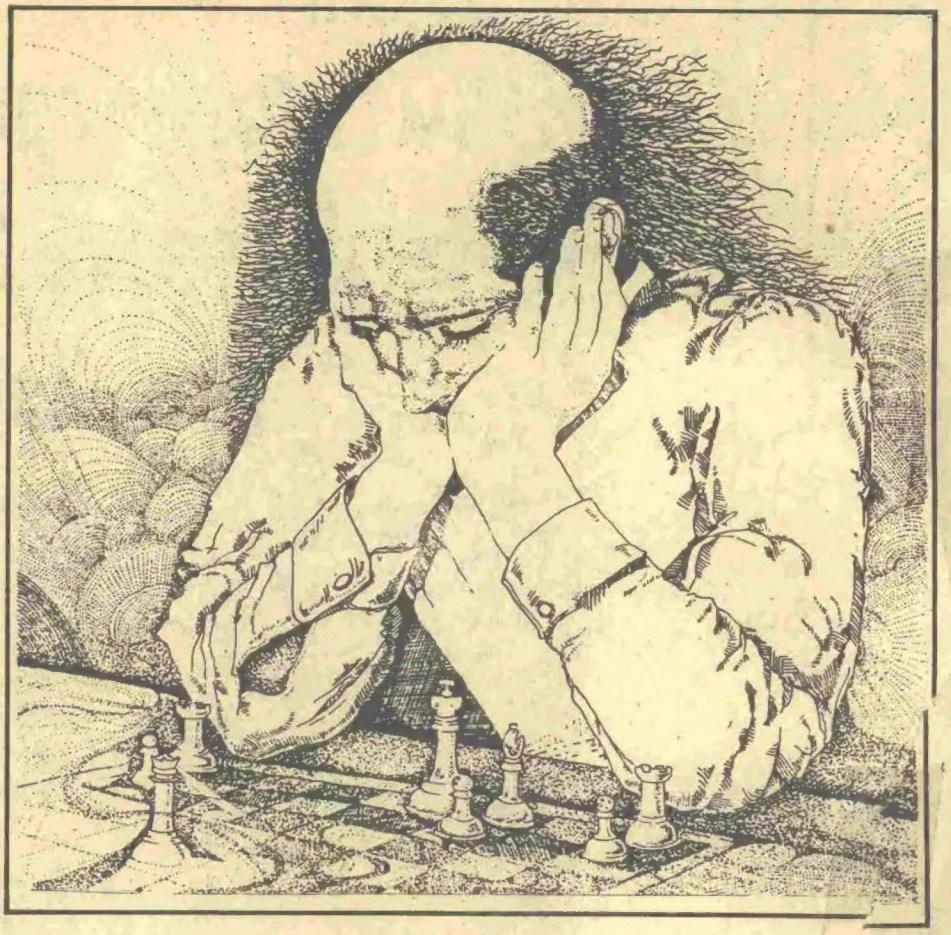



สหัส พรหมสิทธิ์

*นี่คือความเพียรพยายามของมนุษย์ในการคิดค้นสรรค์สร้างโปรแกรมที่จะใส่ "สมอง" ให้กับคอมพิวเตอร์*

วิธีที่จะเข้าถึงอินเทลลิเยนต์โปรแกรมหรือการเขียนคอมพิวเตอร์โปรแกรมเพื่อให้คอมพิวเตอร์แก้ปัญหาต่าง ๆ ได้อย่างชาญฉลาด ในรูปของความสามารถด้านการเรียนรู้และปรับตัวเองได้นั้น คิดว่าน่าจะต้องเริ่มที่โลกของการเขียนโปรแกรมเพื่อสอนให้คอมพิวเตอร์เล่นหมากรุกได้จนเอาชนะคู่ต่อสู้ที่เป็นมนุษย์หรือนักเล่นทั่ว ๆ ไปได้เสียก่อน เพราะวิธีนี้เปรียบเสมือนบันไดที่จะไต่เต้าไปสู่ความเข้าใจในเรื่องของการบรรจุปรีชาญาณความฉลาดเข้าไปในคอมพิวเตอร์ เมื่อประมาณกลางปี ค.ศ.1947 ซึ่งเป็นยุคที่มหาวิทยาลัยชั้นนำต่าง ๆ ของสหรัฐอเมริกาอยากจะได้ชื่อว่าเป็นผู้ร่วมยุคบุกเบิกในการสร้าง

คอมพิวเตอร์ต้นแบบเหมือนที่ฮาร์วาร์ด เอ็มไอที มหาวิทยาลัยเพนซิลเวเนีย โดยจะเริ่มที่มหาวิทยาลัยอิลลินอยส์สถาบันอุดมศึกษาอีกแห่งหนึ่งที่อยากสร้างคอมพิวเตอร์ไว้ใช้ในการศึกษาวิจัยใจแทบขาด แต่กลับได้งบประมาณจากทางมหาวิทยาลัยแค่ 110,000 เหรียญดอลลาร์เท่านั้นเอง ทำอะไรไม่ได้มาก จึงเป็นหน้าที่ของแผนกวิศวกรรมไฟฟ้าของมหาวิทยาลัยแห่งนี้ว่าจะทำอย่างไรจึงจะได้เงินทุนมาอีก และจากจุดนี้เองที่ชื่อของศาสตราจารย์คนหนึ่งในแผนกวิศวกรรมไฟฟ้าของมหาวิทยาลัยอิลลินอยส์เริ่มปรากฏฉายแสงขึ้น นั่นคือ **อาร์เธอร์ แอล แซมมวล** (Arthur L. Samuel) ผู้เล็งเห็นว่าตนควรจะสร้างคอมพิวเตอร์ขนาดเล็กขึ้นมาสักเครื่องหนึ่ง แล้วสอนให้มันเล่นหมากรุกชนิดที่เรียกว่า **เช็คเกอร์** (checkers) เพื่อเรียกร้องความสนใจจากภาครัฐบาลและเอกชนให้สนับสนุนเงินลงขันลงทุนในเรื่องนี้ ผลงานชิ้นแรกของอินเทลลิเยนต์โปรแกรม จึงเริ่ม ณ จุดนี้



## เส้นทางสู่อัจฉริยะโปรแกรม

แซมมวลนั้นเลือกที่จะป้อนโปรแกรมการเล่นเช็คเกอร์เข้าคอมพิวเตอร์ในยุคบุกเบิก เพราะเชื่อว่ากฎเกณฑ์ต่าง ๆ ในการเล่นหมากแบบเช็คเกอร์นั้นไม่ซับซ้อนเหมือนหมากรุกหรือ chess เช็คเกอร์เหมาะสำหรับหน่วยความจำที่แสนจะจำกัดของคอมพิวเตอร์ในสมัยนั้น สรุปแล้วเกมเช็คเกอร์จึงเป็นลูกน้องของ chess อีกทีหนึ่ง ซึ่งการเขียนโปรแกรมสำหรับ chess ในช่วงต่อมานับว่าเป็นหนทางสำคัญอันหนึ่งที่จะนำไปสู่วิชาปัญญาประดิษฐ์ (Artificial Intelligence หรือ AI) อย่างไรก็ตามการศึกษาค้นคว้าของแซมมวลเพื่อเรียนรู้การเขียนโปรแกรมให้คล่องแคล่วตลอดจนพยายามเข้าใจกฎเกณฑ์การเล่นเช็คเกอร์อย่างถ่องแท้ในยามว่างนั้น จัดได้ว่าเป็นอีกเส้นทางหนึ่งที่นำไปสู่ AI ได้เหมือนกัน นั่นคือการสอนคอมพิวเตอร์ให้สามารถเรียนรู้จากประสบการณ์ได้สำเร็จเป็นรายแรก ๆ ในรูป machine learning แม้ว่าเขาจะต้องทำหน้าที่ทั้งอาจารย์และนักค้นคว้า ใน Electron Devices Laboratory ของมหาวิทยาลัยอิลลินอยส์ก็ตาม แต่เขายังสละเวลาส่วนหนึ่งทุ่มเทให้กับการเขียนโปรแกรมสำหรับเช็คเกอร์ แซมมวลจะทำไปพร้อม ๆ กับหน้าที่ที่เขาได้รับมอบหมายจากทางมหาวิทยาลัยให้เป็นหนึ่งในผู้ออกแบบส่วนประกอบที่สำคัญ ๆ สำหรับคอมพิวเตอร์เครื่องแรกของมหาวิทยาลัยแห่งนี้ ผู้เขียนขอรวบรัดเข้าสู่ช่วงหัวเลี้ยวหัวต่อในชีวิตของอาจารย์แซมมวลเกี่ยวกับโปรแกรมเล่นเช็คเกอร์เสียที เพราะเป็นเรื่องน่าสนใจมากในแง่ที่ว่าเขากำลังจะละทิ้งชีวิตในมหาวิทยาลัยเพื่อก้าวไปสู่อีกโลกหนึ่งที่เปิดโอกาสให้เขาได้ขลุกอยู่กับคอมพิวเตอร์แทบทั้งวัน จึงไม่มีอะไรในขณะนั้นที่จะเหมาะเท่ากับการได้ทำงานกับบริษัทผู้สร้างและขายคอมพิวเตอร์ยักษ์ใหญ่ของโลก เช่น ไอบีเอ็ม

แซมมวลเริ่มทำงานกับไอบีเอ็มเมื่อปี 1949 เนื่องจากเล็งเห็นว่าเขามีโอกาสคลุกคลีกับคอมพิวเตอร์มากกว่าที่เดิมซึ่งยังไม่มีอะไรเป็นตัวเป็นตน งานวิจัยของห้องวิจัยไอบีเอ็มที่ปุ๊กคิ๊บซี่ (Poughkeepsie) ในนิวยอร์คจึงน่าจะคู่ควรกับแซมมวลที่สุด และทันทีที่เข้าไปทำงานกับไอบีเอ็มเขาก็มีโอกาสออกแบบคอมพิวเตอร์สำหรับใช้ในด้านธุรกิจการค้าเครื่องแรก ขณะเดียวกันก็พัฒนาโปรแกรมเล่นเช็คเกอร์ (checkers playing program) พร้อมกันไปด้วยในยามว่าง ซึ่งโปรแกรมนี้ก็ชักจะยาวและซับ-

**เมื่อปี 1950 Arthur L.Samuel แห่งไอบีเอ็มเล่นเช็คเกอร์กับคอมพิวเตอร์ขนาดใหญ่ที่เรียกว่า IBM701 แต่ปัจจุบันเครื่องเล็ก ๆ แบบไมโครก็เล่นหมากรุกได้อย่างสบาย แถมเก่งเสียด้วย**

ซ้อนขึ้นทุกที จึงเหมาะเหลือเกินที่จะนำมาทดสอบกับ IBM 701 อันเป็นคอมพิวเตอร์ทางการค้าเครื่องแรกที่เผยโฉมออกมาเมื่อปี 1952 ซึ่งสามารถใช้งานได้ทั่วไปและเก็บโปรแกรมได้ แซมมวลใช้เวลาช่วงกลางคืนเล่นเช็คเกอร์เพื่อตรวจหาที่ผิดของโปรแกรมพร้อมทั้งปรับปรุงไปเรื่อย ๆ บางทีเขาต้องใช้คอมพิวเตอร์ถึงสามตัวให้ทำงานพร้อมกันเพื่อแก้ไขปัญหาหมากรุกของเขา เท่ากับเป็นการเตรียมพร้อมเพื่อให้คอมพิวเตอร์แบบ 701 ก้าวสู่โลกของการใช้งานจริง ๆ ได้โดยไม่มีปัญหา การทดสอบคอมพิวเตอร์ตามวิธีของแซมมวลจึงเปรียบเสมือนการฝึกฝนให้พร้อมที่จะทำงานด้านการคำนวณหรือประมวลผลอันซับซ้อนมาก ๆ ในเวลาใช้งานได้เป็นอย่างดี แต่ทางไอบีเอ็มก็ต้องระวังไม่ให้ข่าวการใช้คอมพิวเตอร์เป็นเครื่องเล่นหมากรุกหรือเช็คเกอร์นี้แพร่ออกไปในหมู่ผู้ใช้เครื่องเพราะเกรงว่าจะเกิดการเข้าใจผิดต่าง ๆ อันอาจจะก่อผลเสียทางการค้า ซึ่งไอบีเอ็มเป็นห่วงเรื่องนี้มากที่สุด แต่จนแล้วจนรอดข่าวก็แพร่สะพัดกระจายไปถึงยุโรป นับเป็นผลดีต่อแซมมวลเองเพราะหลังจากปี 1952 เป็นต้นไป เขาได้มีโอกาสเดินทางไปยุโรปเพื่อสอนวิชาคอมพิวเตอร์และการใช้เครื่อง 701 พร้อมกันไปด้วย กิตติศัพท์ความเป็นนักเขียนโปรแกรมเล่นเช็คเกอร์ช่วยส่งเสริมให้เขามีโอกาสย่างก้าวเข้าไปเยี่ยมชมห้องทดลองทางคอมพิวเตอร์ของหลาย ๆ ประเทศได้อย่างสบาย แถมยังเป็นประโยชน์กับไอบีเอ็มในเวลาต่อมาด้วย แซมมวลดูงานในยุโรปราว 4 ปีก็กลับสหรัฐฯ ปี 1956 จัดว่าเป็นปีที่สำคัญมากในประวัติศาสตร์คอมพิวเตอร์ เพราะในปีนั้นวิทยาลัยดาร์ตมัธ (Dartmouth) ได้จัดการประชุมทาง AI ขึ้นเป็นครั้งแรกของโลก ซึ่งแซมมวลก็เข้าร่วมประชุมเช่นเดียวกัน และได้ให้ข้อคิดว่าไม่ควรหวังอะไรจาก AI ให้มากนักในระยะแรก ๆ อย่างไรก็ตามการประชุมครั้งนี้ถือเป็นย่างก้าวสำคัญที่ทำให้ AI รุดหน้ามาเรื่อยจวบจนปัจจุบัน

## เจาะลึกโปรแกรมเช็คเกอร์

คอมพิวเตอร์โปรแกรมที่เล่นเช็คเกอร์ของแซมมวลในระยะแรกนั้นสามารถเล่นหมากรุกแบบนี้โดยอาศัยการท่องจำ (rote learning) เท่านั้น คือจำลูกเดียวเสร็จแล้วจึงเอาสิ่งที่จดจำไว้มาแข่งขันกับคน ภายในโปรแกรมนั้นแซมมวลจะเขียนกฎในการเล่นเช็คเกอร์เข้าไป และสอนให้คอมพิวเตอร์สามารถดูหมากไว้ล่วงหน้าหลาย ๆ ตา (moves) ในขณะเดียวกันเขาก็ให้คะแนนเป็นตัวเลขสำหรับหมากแต่ละตำแหน่งว่า ถ้าเคลื่อนไปทางนั้นผลที่ตามมาจะเป็นเช่นใด หรือเคลื่อนไปทางนี้แล้วจะเกิดอะไรขึ้น การตัดสินใจของคอมพิวเตอร์สำหรับการเล่นเช็คเกอร์ในช่วงแรก ๆ จึงอาศัย "คะแนน" เป็นเครื่องวินิจฉัย ขั้นต่อมาแซมมวลก็เริ่มป้อนประสบการณ์การหาทางลัดแบบง่าย ๆ เข้าไปในโปรแกรมเพื่อให้คอมพิวเตอร์เล่นเช็คเกอร์ได้เหมือนคนที่เพิ่งหัดเล่นใหม่ ๆ ซึ่งยังไม่เก่งนัก อย่างมากก็พอ ๆ กับตัวของนายแซมมวล เพราะเขาไม่ค่อยถนัดเกมแบบนี้เท่าใดนัก แต่ทำไปดังเหตุผลที่ได้กล่าวไว้แล้วในตอนเริ่มเรื่อง โปรแกรมเช็คเกอร์ในช่วงนี้ของแซมมวลจึงเป็นเสมือนระบบผู้เชี่ยวชาญ (expert system) ที่แพร่หลายในปัจจุบันนี้ ในแง่ความเก่งกาจของโปรแกรมนั้นจะเทียบเท่ากับคนที่ป้อนกฎเกณฑ์ต่าง ๆ หรือความรู้เข้าไป และจะไม่เก่งไปกว่านี้ เพราะยังปรับตัวเองหรือเรียนรู้อะไร ๆ เองไม่ได้เหมือนคน

ต่อมาแซมมวลได้ปรับปรุงโปรแกรมของเขาต่อไปจากโปรแกรมที่ทำหน้าที่เหมือนระบบผู้เชี่ยวชาญที่แพร่หลายอยู่ในขณะนี้ให้ก้าวหน้าต่อไปอีกขั้นหนึ่งด้วยการดัดแปลงให้โปรแกรมเรียนรู้จากประสบการณ์ได้ (learning by generalization) ต้องพึ่งพาการป้อนเงื่อนไขต่าง ๆ เข้าไปเพื่อให้โปรแกรมสามารถจำและเดินหมากโดยอาศัยสิ่งที่ได้เรียนรู้จากประสบการณ์ ส่วนการเลือกว่าต่อไปจะเดินหมากรุกอย่างไรนั้นโปรแกรมจะดูสถิติในอดีตว่าตัวเลขของความสำเร็จและความล้มเหลวแสดงไว้เป็นเช่นใด อันเป็นการรุดหน้าไปอีกก้าวหนึ่งในการสอนให้คอมพิวเตอร์รู้จักเรียนจากประสบการณ์ในอดีต จนมาถึงจุดที่โปรแกรมนี้เคยเล่นเช็คเกอร์ชนะนายแซมมวลมาแล้ว ตอนนั้นอยู่ในราวปี 1962 ครบกำหนด 15 ปีที่เขาค้นคว้า

เรื่องนี้พอดี โปรแกรมของแซมมวลเคยชนะนักเล่นระดับแชมป์ของมลรัฐ เช่น กรณีของ Robert W.Nealey แห่งรัฐคอนเนคติกัต ผู้พลาดท่าพ่ายแพ้แก่คอมพิวเตอร์ในตอน end game แต่โปรแกรมของแซมมวลก็เป็นเพียงจุดหนึ่งในโลกของการเล่นหมากรุกด้วยคอมพิวเตอร์ เพราะนับตั้งแต่นั้นมาโลกของวิทยาการด้านนี้ได้พัฒนาอย่างรวดเร็ว โดยเริ่มต้นจากการเขียนโปรแกรมเล่นเช็คเกอร์มาเป็นการเขียนโปรแกรมและสร้างฮาร์ดแวร์ไว้สำหรับเล่นหมากรุกที่เรียกว่า chess โดยเฉพาะ

## ขอแนะนำให้รู้จักกับแชมป์หมากรุก "BELLE"

คราวนี้ผู้เขียนจะพาไปสัมผัสกับโลกของการเขียนโปรแกรมและการสร้างฮาร์ดแวร์สำหรับเล่นหมากรุกที่เบลล์แล็บ เพื่อให้เห็นภาพที่ชัดเจนว่าหลังจากปี 1962 ซึ่งเป็นปีที่โปรแกรมของนายแซมมวลขยับขึ้นไปถึงจุดที่เกือบอิ่มตัวนั้นในช่วงราว 10 ปี ต่อมาวิทยาการในด้านการสอนคอมพิวเตอร์ให้เล่นหมากรุกที่เรียกว่า chess นั้นได้ก้าวรุดหน้าไปไกลมาก เช่น กรณีของโจ คอนดอน (Joe Condon) แห่งเบลล์แล็บที่ร่วมกับนักโปรแกรมมือฉมังเคน ทอมป์สัน (Ken Thompson) สร้าง BELLE ขึ้นมา ซึ่งการสอนคอมพิวเตอร์ให้เล่นหมากรุกในกรณีของโจ คอนดอนนี้ก็เหมือนกับรายอื่น ๆ คือมักจะเริ่มต้นที่ซอฟต์แวร์ด้วยการเขียนโปรแกรมเพื่อสอนให้คอมพิวเตอร์แบบที่ใช้งานทั่ว ๆ ไป (general purpose computer) ให้เล่นหมากรุก ครั้นพอเห็นว่าวิธีนี้จะไปไม่รอดในแง่ของการเอาชนะแชมป์ได้ยากก็ต้องหันไปพึ่งฮาร์ดแวร์ที่สร้างขึ้นมาเพื่อเล่นหมากรุกโดยเฉพาะ

โปรแกรมเล่นหมากรุกที่เบลล์แล็บเขียนเป็นซอฟต์แวร์ครั้งแรกเมื่อปี 1973 โดยฝีมือของโจ คอนดอนและเคน ทอมป์สัน แล้วนำไปวิ่งกับคอมพิวเตอร์มาตรฐานที่ใช้งานทั่ว ๆ ไป ซึ่งในตอนแรกได้ผลออกมาว่า สามารถหาตำแหน่งเดินหมากได้ 200 ตำแหน่งต่อวินาที ซึ่งยังไม่เร็วเท่าที่ควร โจและเพื่อนร่วมงานจึงหาแนวทางใหม่เนื่องจากคิดไตร่ตรองแล้วว่าจะพึ่งคอมพิวเตอร์แบบมาตรฐานต่อไปอีกไม่ได้ เพราะเป็นฮาร์ดแวร์ทั่วไปที่ไม่เหมาะกับการเล่นหมากรุก เป็นเหตุให้ต้องมีการสร้างคอมพิวเตอร์ขึ้นมาเอง ทั้งสองคนใช้เวลาออกแบบราว 4 ปี คอมพิวเตอร์เล่นหมากรุกขนานแท้เครื่องแรกได้ถือกำเนิดขึ้น (ราวปี 1977) ส่วนประกอบสำคัญภายในนั้นก็คือชิพ (chip) ที่นำมาใช้จำนวน 25 ตัวสำหรับแบบแรก แต่ยังไม่มีท่าทีว่าคอมพิวเตอร์ที่ชื่อว่า BELLE จะคิดอะไรได้เร็วเท่าที่ควร จึงต้องมีการปรับปรุงฮาร์ดแวร์โดยใช้ชิพถึง 325 ตัว ซึ่งสามารถคำนวณได้ 5,000 ตำแหน่งต่อวินาที นั่นเป็นเหตุการณ์เมื่อปี 1980 ต่อมาได้มีการเพิ่มชิพเข้าไปอีกเป็น 1,700 ตัวและคำนวณได้ 160,000 ตำแหน่งต่อวินาทีจัดเป็นแบบล่าสุด คำว่า "BELLE" จึงเป็นทั้งชื่อของฮาร์ดแวร์และซอฟต์แวร์หรืออาจเรียกรวม ๆ ว่าเป็นคอมพิวเตอร์เฉพาะกิจที่มีไว้เล่นหมากรุกโดยเฉพาะก็ได้ จนในที่สุด BELLE ได้รับขนานนามว่าเป็นแชมป์โลกสำหรับหมากรุกที่ 100 มิลลิวินาที ซึ่งหมายความว่าคู่ต่อสู้คิดหมากได้วินาทีละ 10 ตำแหน่งก็จะต้องพ่ายแพ้ BELLE อย่างไม่ต้องสงสัย ดังเช่นการแข่งขัน speed chess ที่จะต้องเล่นเกมให้จบภายใน 5 นาทีในการแข่งที่เรียกว่า United States National Open เมื่อปี 1982 นั้น BELLE ติดอันดับสองในด้าน speed chess

เมื่อพูดถึงการเล่นหมากรุกขนานแท้แล้วก็ต้องพูดถึงการจัดอันดับให้คะแนนของหลักสากลเสียก่อนว่า เขากำหนดความเก่ง (strength) กันอย่างไร ซึ่งการให้คะแนนในรูปของ numerical rating นี้เขาจะดูว่า ใครแข่งกับใคร และแข่งด้วยหลักเกณฑ์ใด ผลปรากฏออกมาว่านักเล่นที่เป็นมนุษย์ (ไม่ใช่คอมพิวเตอร์) เช่น ในสหรัฐฯจะมีคะแนนเฉลี่ยราว 1,400 rating สถิติสูงสุดเท่าที่เคยมีมา จนถึงช่วงปี 1984 คะแนนเฉลี่ยกลับสูงเป็น 2,780 โดยบ๊อบบี้ ฟิสเชอร์ (Bobby Fischer) ส่วน Anatoly Karpov นั้นมีค่าตัวเท่ากับ 2,720

นักเขียนโปรแกรมและผู้สร้างฮาร์ดแวร์ BELLE ไว้สำหรับเล่นหมากรุก ผู้มีนามว่า Joe Condon แห่งเบลล์แล็บ ดังจะเห็นคอมพิวเตอร์ที่ตั้งอยู่ข้าง ๆ ขณะเล่นหมากรุก (chess) ซึ่งคอมพิวเตอร์เฉพาะกิจที่เห็นนี้สามารถวิเคราะห์เพื่อวางแผนการเล่นได้ถึง 6,000 ตาต่อวินาที

grand master rating จะเริ่มที่ 2,450 ส่วนคอมพิวเตอร์สำหรับเล่นหมากรุกที่มีขายเมื่อ 4-5 ปีก่อนจะได้ rating เท่ากับ 1,700 คะแนน ส่วน BELLE เมื่อ 4-5 ปีมาแล้วจะมี rating เท่ากับ 2,200 หมายถึงว่าทั้งคอมพิวเตอร์และซอฟต์แวร์ดังกล่าวจะสามารถเอาชนะนักเล่นทั่วโลกได้ถึงร้อยละ 99 และปี 1983 เป็นปีที่ BELLE ได้ master rating อย่างเป็นทางการจาก International Chess Federation ที่กล่าวมาทั้งหมดเป็นเหตุการณ์เพียงช่วงหนึ่งของวิวัฒนาการการเขียนโปรแกรมและการสร้างฮาร์ดแวร์เพื่อเล่นหมากรุก ซึ่งผู้เขียนเห็นว่ามีอะไรหลาย ๆ อย่างที่เกี่ยวข้องกับโปรแกรมประเภทชาญฉลาด จึงขอวกกลับเข้าเรื่องอันเป็นหัวใจสำคัญของบทความนี้ต่อไปในแง่ของความสัมพันธ์ระหว่างสิ่งที่ได้กล่าวมาแล้วทั้งหมดกับวิทยาการทางด้าน AI

## ต้นกำเนิดของโปรแกรมการเรียนรู้

ความพยายามของนักคอมพิวเตอร์ศาสตร์และนัก AI ที่จะหาทางเขียนโปรแกรมให้มีปรีชาญาณในตัวเองนั้นยังคงดำเนินเรื่อยมานับตั้งแต่ Arthur L. Samuel แห่งไอบีเอ็มได้บุกเบิกงานด้านนี้ด้วยการเขียนโปรแกรมสำหรับเล่นเช็คเกอร์เป็นรายแรก ๆ เมื่อปี 1949 แต่งานค้นคว้าที่น่าสนใจที่สุดชิ้นหนึ่งที่จะเข้ามาเกี่ยวข้องกับภาคปฏิบัติเพื่อใช้งานในโลกแห่งความเป็นจริงนั้นดูเหมือนจะเป็นการริเริ่มของนัก AI ผู้มีนามว่าริสซาร์ด ไมแคลสกี้ (Ryszard Michalski) แห่งมหาวิทยาลัยอิลลินอยส์ ผู้ที่สร้างโปรแกรมที่เรียกว่า AQ11 ขึ้นมาในปี 1976 การทำงานของ AQ11 เริ่มต้นด้วยการป้อนข้อมูลแสดงโรคพืชที่เกิดกับต้นถั่วหรือ soybean โดยรวมอาการต่าง ๆ ของโรคเข้าไว้ด้วย เช่น สีและอาการของใบและกิ่งก้านต่าง ๆ แล้วไปสิ้นสุดที่ความเห็นของผู้เชี่ยวชาญในด้านโรคพืชว่าได้วินิจฉัยโรคของต้นถั่วจำนวน 290 ต้นไว้อย่างใดบ้าง จากนั้นก็นำข้อมูลดังกล่าวไปประมวลผล ในตอนนี้เองที่กลไกในการหาเหตุผล ของ AQ11 จะสอนให้ตัวโปรแกรมรู้จักเรียนรู้และสร้าง "กฎ" จากข้อมูลทั้งหมดที่ป้อนเข้าไป (กฎนี้จะอยู่ในรูป

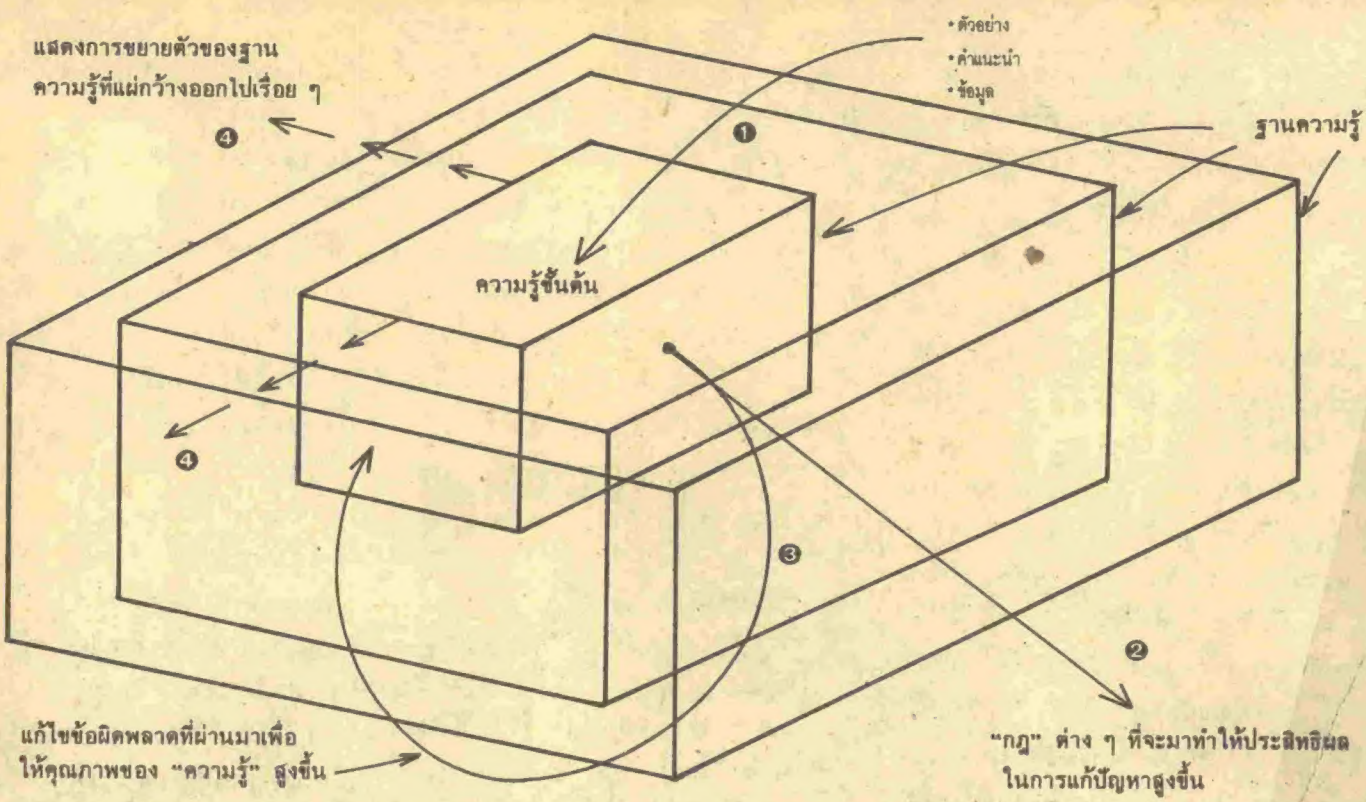
**Arthur L.Samuel แห่งไอบีเอ็ม ผู้บุกเบิกงานด้าน AI ใช้วิธีการเล่นโปรแกรมเช็คเกอร์ เพื่อให้คอมพิวเตอร์สามารถเรียนรู้ได้จากความผิดพลาดของมันเอง**

ของ if ...then... ที่นิยมใช้สร้างฐานความรู้ของระบบผู้เชี่ยวชาญในปัจจุบัน จัดเป็นโปรแกรมในด้าน AI ชิ้นแรก ๆ ที่เป็นระบบผู้เชี่ยวชาญสามารถสอนตัวเองได้ในลักษณะของการเรียนรู้จากตัวอย่างเป็นจำนวนมากที่ป้อนเข้าไป ซึ่งในกรณีนี้ก็คืออาการของโรคในต้นถั่วเป็นจำนวน 290 ต้นนั่นเอง แต่ในทางปฏิบัตินั้นโรคที่เกิดกับต้นถั่วที่มักพบบ่อย ๆ จะมีเพียงสิบกว่าชนิดเท่านั้น

การทดลองขั้นต่อไปของไมแคลสกี้ก็คือ ลองสร้างระบบผู้เชี่ยวชาญอีกระบบหนึ่งขึ้นมาแข่งความแม่นยำกับ AQ11 ในการวิเคราะห์โรคพืชของต้นถั่วอยู่ 15 ชนิดด้วยกัน ระบบผู้เชี่ยวชาญนี้อาศัยความรู้ในรูปของกฎที่เป็น if...then...ที่ได้มาจากนักพยาธิวิทยาโรคพืช (plants pathologists) ชั้นนำในสมัยนั้น จุดประสงค์เพื่อต้องการดูว่าความแม่นยำในการวินิจฉัยโรคพืชของ AQ11 ที่สร้างกฎได้เองกับระบบผู้เชี่ยวชาญแบบใหม่ที่สร้างกฎโดยอาศัยความเห็นของผู้เชี่ยวชาญที่เป็นมนุษย์ว่าใครจะแม่นกว่ากัน จากนั้นก็ป้อนอาการของโรคในต้นถั่วเป็นจำนวน 340 ต้นเข้าคอมพิวเตอร์แล้วสั่งให้ประมวลผลโดยแยกกันเป็นสองครั้ง ครั้งแรกเป็นการใช้โปรแกรมที่เรียกว่า AQ11 ที่สร้างกฎได้เองเมื่อเราป้อนตัวอย่างแสดงอาการของโรคเข้าไปจากต้นถั่ว 340 ต้น ผลของการคำนวณปรากฏว่า AQ11 วินิจฉัยโรคได้ถูกต้องถึงร้อยละ 97.6 ส่วนระบบผู้เชี่ยวชาญใหม่ที่สร้างกฎเองไม่ได้นั้นให้ความแม่นยำได้เพียงร้อยละ 71.8 เท่านั้น ย่อมแสดงว่า AQ11 เป็นโปรแกรมหรืออินเทลลิเยนต์โปรแกรมที่เรียนรู้ด้วยตัวเองจากตัวอย่างที่ป้อนเข้าไปเป็นจำนวนมากนั้นจะแม่นยำในการวินิจฉัยเพื่อหาสาเหตุหรืออาการต่าง ๆ ได้เป็นอย่างดี จึงเหมาะที่จะใช้กับงานในบางประเภท เช่น แพทยศาสตร์ ฟิสิกส์ หรือแม้กระทั่งการเล่นหมากรุก ซึ่งเป็นอีกก้าวหนึ่งในการสร้างคอมพิวเตอร์โปรแกรมที่สามารถเรียนรู้จากตัวอย่างจำนวนมากที่ป้อนเข้าไปเป็นข้อมูล

## พบกับความน่าทึ่งในการเรียนรู้ของเครื่อง

การเรียนรู้ของเครื่อง (machine learning) เป็นกระบวนการในการเพิ่มพูนความรู้ให้แก่คอมพิวเตอร์ รวมทั้งการทำให้คอมพิวเตอร์ปรับเปลี่ยนพฤติกรรมของตัวเองได้อันเนื่องมาจากประสบการณ์และผลของการคิดการกระทำต่าง ๆ ที่ผ่านมา ซึ่งในการนี้มักมีคำว่าฮิวริสติก (heuristics) เข้ามาเกี่ยวข้องบ่อยครั้ง อันหมายถึงการใช้วิจารณญาณที่ดี เพื่อทำให้การแก้ปัญหาต่าง ๆ นั้นสามารถทำได้อย่างรวบรัดมากขึ้น เพราะฮิวริสติกจะให้สัมฤทธิ์ผลในรูปต่าง ๆ ที่เกิดจากประสบการณ์แฝงซึ่งอยู่ภายในตัวเรา




**รูปที่ 1 โมเดลของปฏิกิริยาระหว่างความรู้ที่มีอยู่ในขั้นต้นกับอินพุตและเอาต์พุตในรูปแบบต่าง ๆ**

แมชีน เลิร์นนิ่งนั้นเป็นผลมาจากปฏิกิริยาระหว่างความรู้ที่มีอยู่ในฐานข้อมูลขั้นต้นกับอินพุตและเอาต์พุตในรูปแบบต่าง ๆ ดังรูปที่ 1 ซึ่งจะเริ่มด้วยอินพุตที่แสดงด้วยลูกศรเลขที่ 1 อาจอยู่ในรูปของคำแนะนำจากผู้ใช้ ตัวอย่างหรือข้อมูลของคอมพิวเตอร์โปรแกรมหรือเลิร์นนิ่งโปรแกรมจะต้องตีความและนำไปใช้ประโยชน์ในการแก้ปัญหาให้ได้ผล ขณะที่โปรแกรมทำการประมวลผลจากคำแนะนำ ซึ่งข้อมูลดังกล่าวจะถือโอกาสแก้ไขสิ่งที่อยู่ในรูปของความจริง (facts) และสูตรจัดเจนหรือฮิวริสติก (heuristics) ต่าง ๆ ภายในฐานความรู้ (knowledge) พร้อมกันไป ซึ่งการแก้ไขปรับปรุงนี้จะแสดงผลออกมาทางเอาต์พุตหรือผลของการแก้ปัญหาดังแสดงด้วยลูกศรหมายเลข 2 คราวนี้จะขอวกกลับไปอธิบายถึงการตีความที่กล่าวในตอนแรกสักเล็กน้อย คือเมื่อเลิร์นนิ่งโปรแกรมทำการตีความข้อมูลคำแนะนำหรือ/และตัวอย่างต่าง ๆ แล้วจะต้องสามารถสร้างกฎเกณฑ์ต่าง ๆ ขึ้นมาได้เอง ในกรณีที่สร้างกฎเกณฑ์ที่ตายตัวไม่ได้ก็จะต้องหาแนวความคิดหรือ concepts กว้าง ๆ ออกมาให้ได้ เหมือนเช่นโปรแกรมที่เรียกว่า AQ11 ที่กล่าวไว้ในหัวข้อก่อนหน้านี้ ผลของการแก้ปัญหาที่ได้จากคอมพิวเตอร์ตามลูกศรเลขที่ 2 นั้นอาจหมายถึงการแก้ปัญหาได้อย่างแม่นยำขึ้น หรือหมายถึงการหาทางออกต่าง ๆ ที่ทำให้ค่าใช้จ่ายต่ำลง ในบางกรณีอาจหมายถึงความสามารถของคอมพิวเตอร์ในการขยายขอบเขตของการแก้ปัญหาต่าง ๆ ได้มากรูปมากแบบขึ้นด้วยการใช้โมเดลพิเศษบางอัน จนถึงขั้นที่คอมพิวเตอร์สามารถแปลกฎเกณฑ์ต่าง ๆ ที่เก็บไว้ในฐานความรู้ให้ง่ายเข้า เพื่อประหยัดเนื้อที่ของหน่วยความจำ ก็เป็นสิ่งที่ไม่เหลือวิสัยในอนาคต สุดท้ายก็จะมี feedback loop ที่แสดงด้วยลูกศรเลขที่ 3 จะคอยดูผลของการทำงานของโปรแกรมด้วยการเปรียบเทียบผลที่ได้กับเป้าหมายที่กำหนดไว้ พร้อมทั้งหาข้อผิดพลาดและให้คำแนะนำว่าควรจะแก้ไขอย่างไรต่อไปดี ซึ่งการแก้ไขปรับปรุงต่าง ๆ นี้จะสะท้อนออกมาให้เห็นในการที่ฐานความรู้จะขยายตัวและมีขนาดใหญ่ขึ้นเรื่อย ๆ ดังลูกศรเลขที่ 4 ทั้งหมดนี้เป็นแนวความคิดในเรื่องของแมชีน เลิร์นนิ่งที่ต้องทำงานโดยอาศัยเลิร์นนิ่งโปรแกรมควบคู่กันไป

แนวความคิดที่แสดงไว้ในรูปที่ 1 นั้นประกอบด้วยส่วนสำคัญ ๆ ที่โมเดลแบบต่าง ๆ ในด้านแมชีน เลิร์นนิ่งมักจะใช้เป็นแบบอย่างอยู่บ่อย ๆ ในการสอนให้คอมพิวเตอร์เรียนรู้เป็น ไม่ว่าจะเป็นการเรียนรู้จากตัวอย่าง เรียนรู้จากความคล้ายคลึง เรียนรู้จากการค้นหา หรือเรียนรู้จากการสอน ศูนย์กลางของโมเดลที่แสดงในรูปที่ 1 ก็คือสิ่งที่เป็นแก่นที่เรียกว่าความรู้ขั้นต้น (initial knowledge) ที่แสดงด้วยสี่เหลี่ยมลูกบาศก์ด้านในสุด การสร้างความรู้ขั้นต้น

ในกรณีนี้ก็เหมือนกับระบบผู้เชี่ยวชาญในแง่ที่ว่า ผู้เขียนโปรแกรมจะต้องกำหนดความรู้ดังกล่าวอันประกอบด้วยฮิวริสติก (heuristics) ต่าง ๆ ที่มักจะอยู่ในรูปของ if ... then ... พร้อมด้วยสิ่งที่เรียกว่าความจริง (facts) นอกจากนี้ภายในสิ่งที่เรียกว่าความรู้ชั้นต้นยังต้องมีโปรแกรมหรือ coded instructions ที่จะมาบริหารจัดการกับข้อมูลต่าง ๆ รวมทั้งส่วนที่เป็น metaknowledge ซึ่งหมายถึงการที่โปรแกรมจะสามารถหยั่งรู้ความสามารถ ตลอดจนขีดจำกัดของตัวมันเองได้ตลอดเวลา เลิร์นนิ่งโปรแกรมจึงก้าวล้ำนำหน้าคอมพิวเตอร์โปรแกรมทั่ว ๆ ไป รวมทั้งระบบผู้เชี่ยวชาญที่ใช้กันแพร่หลายในขณะนี้ ในแง่ที่ว่าโปรแกรมโดยทั่วไปและระบบผู้เชี่ยวชาญส่วนใหญ่จะทำงานโดยอาศัยกฎเกณฑ์ที่ผู้เขียนโปรแกรมเป็นผู้กำหนดขึ้นมา แล้วก็จะเป็นอยู่อย่างนั้นจนกว่าจะมีใครเข้าไปแก้ไขหรือเพิ่มเติมกฎเกณฑ์ดังกล่าวเพื่อเป็นการปรับปรุงให้ดีขึ้น เพราะมันปรับปรุงตัวเองหรือเรียนรู้เพิ่มเติมด้วยตัวเองไม่ได้ จึงแตกต่างจากเลิร์นนิ่งโปรแกรมตรงข้อนี้ ประโยชน์ใช้สอยของเลิร์นนิ่งโปรแกรมที่สำคัญ ๆ จึงมีมาก โดยเฉพาะในกรณีที่มนุษย์ต้องอาศัยคอมพิวเตอร์ให้เข้ามาช่วยในการแสวงหาสาเหตุความเป็นไปต่าง ๆ รวมทั้งตัดสินใจแทนมนุษย์ ซึ่งขณะนี้มีผลงานอันเป็นประจักษ์พยานปรากฏออกมาเรื่อย ๆ ว่า เลิร์นนิ่งโปรแกรมได้เข้ามาทำในสิ่งที่ครั้งหนึ่งทำไม่ได้ เช่น เข้ามาช่วยให้การวิจัยทางวิทยาศาสตร์รุดหน้าไปเร็วขึ้น ด้วยความสามารถในการหาร่องรอยที่แสดงความสัมพันธ์ต่าง ๆ ออกมาให้เห็นอย่างเด่นชัดจากตัวเลขข้อมูลจำนวนมากมายก่ายกองที่ไม่มีทีท่าว่าจะมีอะไรสัมพันธ์กันเลย เพราะความสัมพันธ์ของตัวเลขข้อมูลในกรณีนี้จะซ่อนอยู่อย่างลึกมาก ๆ จนแทบเป็นไปไม่ได้ที่จะจำแนกด้วยวิธีธรรมดา สิ่งที่กล่าวมานี้เป็นเพียงตัวอย่างเล็ก ๆ อันหนึ่งเท่านั้น

## โครงการ Cyclopedia ความมหัศจรรย์ในยุค AI

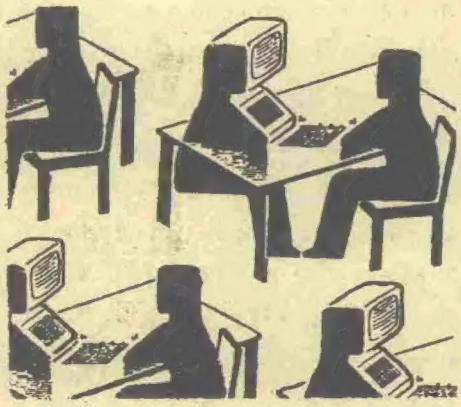
การใช้คอมพิวเตอร์ที่สำคัญมาก ๆ ในยุคต่อไปนั้นจะเป็นไปในรูปของ artificially intelligent machine หรือเลิร์นนิ่งแมชีน ซึ่งประเทศต่าง ๆ ที่เป็นผู้นำทางเทคโนโลยีต่างก็ตระหนักในเรื่องนี้ เช่น การตั้ง MCC (Microelectronics and Computer Technology Corporation) ที่เมืองออสติน รัฐเท็กซัส เมื่อปี 1982 ทั้งนี้โดยมีจุดมุ่งหมายที่จะเร่งรัดเพื่อสร้างคอมพิวเตอร์ที่มีปรีชาญาณความฉลาดให้เกิดเป็นตัวเป็นตนให้ได้สักวัน เพื่อไม่ให้ประเทศอื่น

สามารถทาบรัศมีหรือเข้ามาแก่งแย่งตลาดการค้าในด้านนี้ไปจากตน ผู้ให้การสนับสนุนทางการเงินแก่ MCC ก็คือบริษัทที่สร้างและขายคอมพิวเตอร์รายใหญ่ ๆ ของสหรัฐฯตลอดจนผู้ที่อยู่ในอุตสาหกรรม AI ส่วนหลักเกณฑ์ในการค้นคว้าวิจัยที่ MCC นี้ก็คือการเชิญผู้ทรงคุณวุฒิในด้านคอมพิวเตอร์และ AI จากสถาบันชั้นนำต่าง ๆ ในสหรัฐฯให้มาประจำอยู่ที่นี่ และเปิดโอกาสให้ผู้ทรงคุณวุฒิดังกล่าวทำการค้นคว้าในสาขาที่ถนัด ดังเช่นกรณีของนัก AI นามกระเดื่องที่มีชื่อว่า Douglas B.Lenat ที่พยายามจะใช้เวลาประมาณ 10 ปี เพื่อสอนคอมพิวเตอร์ให้รู้อะไร ๆ เหมือนเด็กอายุ 6 ขวบให้ได้ โดยตั้งชื่อโครงการของเขาว่า "ไซค์" (CYC ย่อมาจาก Cyclopedia)

ศาสตราจารย์ดักลาส บี. เลแนต เคยสอน AI ที่สแตนฟอร์ดและคาร์เนกิเมลลอน เป็นนักวิจัยในด้าน AI ที่ค่อนข้างบ้าบิ่นคนหนึ่งก่อนที่จะมาอยู่ที่ MCC สไตล์การค้นคว้าของอาจารย์ผู้นี้จะยึดหลักของการกระโดดเข้าไปแก้ปัญหาที่เข้ามาท้าทายตนโดยจะไม่เกาะทฤษฎีหรือหลักการจนเกินไป อันจะทำให้การค้นคว้าวิจัยสิ่งแปลก ๆ ใหม่ ๆ ต้องเสียรูปรสไปหมด จึงจัดได้ว่าเขาเป็นนักเสี่ยงตัวฉกาจในแง่ของงานค้นคว้าต่าง ๆ ทางด้าน AI ที่เคยทำมาแล้วในอดีตและกำลังจะทำต่อไปที่ MCC ดังในกรณีของไซค์ แม้คนส่วนหนึ่งในแวดวง AI จะมองเห็นว่าโอกาสที่จะทำไม่สำเร็จมีอยู่หลายเปอร์เซ็นต์ เพราะเป็นงานวิจัยที่เย้ยฟ้าท้าดินอย่างยิ่ง จุดมุ่งหมายของเขาในโครงการไซค์นี้เป็นความหมายที่จะทำให้คอมพิวเตอร์คิดเองเป็นและสามารถเรียนรู้สิ่งใหม่ ๆ เหมือนคน ซึ่งจะต้องใช้โปรแกรมเมอร์มาช่วยงานถึง 35 คน พร้อมกับค่าใช้จ่ายไม่น้อยกว่า 25 ล้านเหรียญสหรัฐฯ โครงการนี้เริ่มมาได้ราว 6 ปีแล้ว ดังนั้นอีกราว 4 ปีจากนี้ก็คงทราบผลว่าจะสำเร็จหรือล้มเหลว ถ้าประสบความสำเร็จก็เท่ากับเป็นการปฏิวัติพลิกแผ่นดินในด้าน AI ทีเดียว อันจะส่งมรรคผลที่เป็นเงินเป็นทองตามมาอีกนับไม่ถ้วนหากมีการจำหน่ายถ่ายทอดเทคโนโลยีเช่นนี้ให้ผู้ซื้อทั่วโลก เช่น ในรูปของผลิตภัณฑ์ที่เรียกว่า automated intelligent companion หรือเพื่อนคู่คิดแบบอัตโนมัติ จึงจะขอเริ่มตรงจุดที่ว่าจะสอนคอมพิวเตอร์ให้

มีปรีชาญาณเป็นของตัวเองได้อย่างไร

ก่อนที่คอมพิวเตอร์จะมีปรีชาญาณเป็นตัวของตัวเองได้นั้นจะต้องมีการป้อน "ความรู้" เข้าไปเป็นจำนวนมากในรูปของฐานข้อมูลต่าง ๆ ที่เกี่ยวข้องเชื่อมโยงซึ่งกันและกันจนเหมือนกับมนุษย์ในแง่ที่ว่าก่อนที่ใครสักคนจะแก้ปัญหาต่าง ๆ ได้นั้นจะต้องเรียนรู้สิ่งต่าง ๆ มากพอสมควร ซึ่งอาจต้องเริ่มจากความรู้ทั่วไปจนถึงความรู้เฉพาะเรื่องหรือเฉพาะสาขา กรณีของไซค์ (CYC) นี้จึงต้องมีการจัดประเภทและหมวดหมู่ของความรู้ที่จะป้อนเข้าไปโดยอาศัยความรู้ทั่วไปที่ป้อนเข้าไปในคอมพิวเตอร์ในชั้นแรก ๆ ให้เป็นพื้นฐานสำหรับรองรับความรู้ขั้นต่อ ๆ ไปที่จะป้อนเข้าไปภายหลัง ความรู้อันเป็นพื้นฐานจึงสำคัญมาก และจะกำหนดขอบเขตว่าจะต้องกว้างและละเอียดแค่ไหนจึงเกิดประโยชน์ ซึ่งขณะนี้ศาสตราจารย์เลแนตนัก AI วัย 40 ได้เล่าว่า ไซค์มีความรู้ขั้นพื้นฐานเกือบสมบูรณ์แล้ว เพราะคอมพิวเตอร์เริ่มจะเรียนรู้สิ่งใหม่ ๆ ด้วยตัวเองโดยอาศัยความรู้เก่าที่เป็นพื้นฐานเป็นเครื่องคิดคืบหน้าหรือเครื่องมือในการสานต่อหรือเป็นอาการที่เรียกว่า semantic convergence ที่สืบเนื่องมาจากฐานข้อมูลของคอมพิวเตอร์ มีแนวความคิดที่เป็นพื้นฐานมากพอที่จะเริ่มหาความรู้ใหม่ออกมาได้เอง ซึ่งในขั้นของการป้อนข้อมูลและความรู้ต่าง ๆ เข้าไปนั้นต้องใช้จำนวนเจ้าหน้าที่ถึงเจ็ดคนเพื่อทำงานนี้โดยเฉพาะ และถ้าทุกสิ่งทุกอย่างเป็นไปตามที่คาดคิดไว้ "ไซค์" ก็จะเป็นองค์กรอันสำคัญที่สามารถเรียนรู้หรือจัดการกับสิ่งที่เป็นข้อมูลความรู้ต่าง ๆ ได้ด้วยตัวเองในรูปของคอมพิวเตอร์ที่มีความชาญฉลาดสูง และพร้อมที่จะเข้าไปเชื่อมโยงกับฐานข้อมูลใหญ่ ๆ ที่กระจายอยู่ตามที่ต่าง ๆ ไม่ว่าจะเป็นสหรัฐฯหรือประเทศต่าง ๆ ทั่วโลก เพื่อจะได้มีโอกาสนำข้อมูลในฐานข้อมูลดังกล่าวมาใช้ประโยชน์ได้เต็มที่โดยอาศัย "สามัญสำนึก" ที่มีอยู่ใน "ไซค์" เป็นหัวใจในการเลือกเฟ้นพิจารณาว่า ปัญหาใดควรจะแก้ไข ด้วยการใช้ฐานข้อมูลจากที่ใดบ้างจึงจะเกิดสัมฤทธิ์ผล ที่น่าสนใจที่สุดก็คือ ไซค์จะทำหน้าที่เป็นศูนย์กลางในการใช้สามัญสำนึก ซึ่งในเชิงการค้าแล้วผู้ที่สนใจจะใช้บริการของไซค์อาจโยงฐานข้อมูลของตนหรือของคนใดคนหนึ่งเข้ากับคอมพิวเตอร์ของไซค์ เพื่อให้เป็นพี่เลี้ยงทางปัญญาในเวลาที่ต้องการจะแก้ปัญหาในเรื่องใดเรื่องหนึ่งที่ต้องอาศัยความรู้และข้อมูลอันกว้างไกล หรือในบางกรณีที่อาศัยเพียงความรู้เฉพาะสาขาก็ทำได้เช่นกัน



## ส่งท้ายกับตัวอย่าง

ผู้เขียนยกตัวอย่างเป็นการส่งท้ายในแง่ของการใช้ประโยชน์จากไซค์ในอนาคตอันใกล้ (ราวปี 1996 ที่คาดว่าไซค์จะเสร็จ) ว่าจะทำอะไรได้บ้าง เช่น โรงพยาบาลที่มีแผนการในการบริหารงานโดยเชื่อมโยงกับไซค์ก็จะสามารถทำการวิเคราะห์วางแผนอะไรได้หลาย ๆ อย่างโดยต่อเนื่องและไม่ต้องพึ่งโปรแกรมเมอร์แต่อย่างใด ในกรณีที่ต้องการแก้ปัญหาในด้านสาธารณภัย อาทิ แผ่นดินไหว ก็อาจทำได้ในทันทีโดยสมองเทียม เช่น ให้ไซค์สกัดรายละเอียดข่าวคราวของแผ่นดินไหว ณ ที่ใดที่หนึ่งจากสำนักข่าวแล้ววิเคราะห์ข่าวดังกล่าวว่าเกิดการบาดเจ็บเสียหายมากน้อยเพียงใด ซึ่งทุกอย่างจะเป็นไปโดยไม่ต้องมีใครเข้าไปแทรกแซงเสร็จแล้วไซค์ก็จะใช้ระบบผู้เชี่ยวชาญอันเป็นเสมือนคอมพิวเตอร์โปรแกรมขนาดย่อม (เมื่อเทียบกับไซค์) ที่มีอยู่แล้วให้ไปทำการแนะนำบุคคลที่เกี่ยวข้องว่าควรจะแก้ปัญหาในการบรรเทาสาธารณภัยอย่างไรต่อไป ส่วนอีกตัวอย่างก็จะอยู่ในรูปที่ว่า ผู้ที่ต้องรับผิดชอบในการพัฒนาระบบผู้เชี่ยวชาญต่าง ๆ ไว้ใช้งานสามารถโยงฐานข้อมูลที่มีความรู้ในสาขาใดสาขาหนึ่งอย่างแคบ ๆ ให้เข้ามารวมกับไซค์ แล้วให้ไซค์ทำหน้าที่ช่วยระบบผู้เชี่ยวชาญนั้น ๆ ในแง่ของการใช้สามัญสำนึก เมื่อเกิดปัญหาแปลก ๆ ใหม่ ๆ ผ่านเข้ามาให้ขบคิดเพื่อแก้ไข คือถ้าจะเปรียบเทียบว่า ระบบผู้เชี่ยวชาญนั้นเป็น "มือ" ไซค์ก็จะเปรียบเสมือน "สมอง" ที่คอยสั่งการไปยังอวัยวะชั้นปลายแถวอีกต่อหนึ่ง ไซค์จึงทำหน้าที่เหมือนซูเปอร์ซิสเต็ม (supersystem) ที่จะเป็นสมองสากลที่ใคร ๆ สามารถให้ความไว้วางใจในการแก้ปัญหาตั้งแต่ขนาดย่อมจนถึงระดับชาติได้เป็นอย่างดี จนถึงจุดที่ว่าไซค์จะก้าวเข้าไปดูแลจัดการใช้งานระบบผู้เชี่ยวชาญขนาดใหญ่ที่แต่ละประเทศมีอยู่แล้ว โดยทำงานเชื่อมโยงร่วมกันระหว่างไซค์กับระบบผู้เชี่ยวชาญในสาขาต่าง ๆ เพื่อร่วมกันแก้ปัญหาระดับสากลให้เป็นที่สัมฤทธิ์ผลในเวลาอันสั้นย่อมมีทางเป็นไปได้มากที่สุดในวันข้างหน้าแน่นอน